# APPENDIX E. SOFTWARE CONFIGURATION CONTROL PROCEDURE

This appendix provides an acceptable approach to ensure that the nuclear criticality safety software system used in support of contractor installation nuclear criticality safety organization(s) will

- provide accurate and reliable results,
- provide rigorous structure to implement software changes, and
- prevent unauthorized changes to the software.

Included in this appendix are the actions and responsibilities for maintaining the quality and integrity of the nuclear criticality safety software system used in support of the contractor installation nuclear criticality safety organization(s). Except when specifically included in a Software Catalog, vendor-supplied systems software, such as operating systems, linkers, compilers, and data base management systems used by the contractor installation, are excluded here and covered by separate configuration control for which the contractor is responsible.

E.1 Specific responsibilities.

E.1.1 Contractor safety organization manager. The contractor safety organization manager

- acts as or appoints a Software System Team Chairperson;
- assumes overall responsibility for the configuration control of the Contractor Nuclear Criticality Safety Software System;
- maintains membership and charter on the Software System Team through coordination with the Contractor Nuclear Criticality Safety Committee (See Form E.7 for charter);
- schedules and coordinates annual surveillance of the software configuration control program;
- requests a surveillance/audit of configuration control for software utilized for nuclear criticality safety computations once every five years;
- maintains a current listing of authorized users as notified by System Administrator;
- distributes pertinent information on the software changes, Software Catalog, validations, and other sources to authorized users as appropriate;
- participates, coordinates, and manages the handling and resolution of Software Revision Reports and Software Nonconformance Reports as prescribed in this plan; and
- maintains hard copy documentation for a retention period consistent with section 2.1.2 for

  - Software Configuration Control Plans,
  - Software Catalogs (Form E.3),

- Software Revision Reports (Form E.1),
- Software Nonconformance Reports (Form E.2),
- Request for User Access (Form E.4),
- Audit and Surveillance Reports, and
- Software System Team Charter and Membership.

E.1.2 Software System Team. The Software System Team

- by majority, determines those development, verification, testing, and record keeping operations to be covered by the Configuration Control Plan and the access controls to be required,

- when a new software system is believed to be ready for use, reviews and approves the Software Catalog for completeness and correct access control,

- develops the requirements for software Verification and Configuration Control Tests, coordinates the performance of the required tests, and approves all new or revised software before production use,

- ensures that documentation has been updated (e.g., Configuration Control Plan, Software Catalog, Access Control, records of Verification and Configuration Control Tests, etc.),

- upon request, assists the "quality organization" and other organizations in performing software appraisals, audits, and surveillances,

- when a change to the software is requested, reviews the Software Change Request, Software Revision Report, Part A (Form E.1), to decide if and when the change should be made and completes Parts B and C, as appropriate,

- reviews Software Nonconformance Reports (Form E.2) and determines and documents resolution by a majority in agreement, and

- develops, implements, and maintains a Software Disaster Plan, as appropriate (Form E.5).

E.1.3 Functional System Manager. The Functional System Manager

- serves as the principal Nuclear Criticality Safety Organization contact to the software user with regard to the content of the software,

- provides notification to installation software users of changes to the software systems, nonconformance reports, specialized machine dependent job control language (JCL) requirements, and current Software Catalog, and, if serving as the lead contact for more than one installation, maintains communication with each installation represented,

- participates in the handling and resolution of Software Revision Reports and Software Nonconformance Reports as prescribed in this plan,

- ensures that a Software Catalog is prepared for each mainframe computer software application, is kept current, and a copy provided to each user,

- verifies correct version of software is transferred into Migration Storage Area from the Development Storage Area and performs or coordinates the Software Verification Tests, and

- upon approval of the change for production use, ensures that the version identification of any departmental procedure or plans that reference the software by version are updated.

E.1.4 System Administrator. The System Administrator

- ensures that master copies of the previous versions of machine executable modules and source code are maintained in the Archive Storage Area, and that a hard copy listing and documentation of the latest version are maintained,

- retains a copy of all Software Revision Report (Form E.1) forms,

- prepares the Software Catalog and sends a copy of each updated catalog to all members of the Software System Team,

- notifies the Software System Team that programming of a requested revision is complete and has been transferred to the Migration Storage Area for Verification Testing,

- checks the Software Revision Reports and supporting documentation for completeness and forwards the report to the Software Developer,

- performs the transfer of software to the Production Storage Area and Archive Storage Area when all proper tests and approvals authorize the transfer,

- verifies and ensures the proper version of the executable code is in the Production Storage Area and the most recently superseded version of the source and executable code is stored in the Archive Storage Area,

- develops, implements, and maintains the Configuration Control testing of the software production version and maintains appropriate documentation of testing,

- develops, implements, and maintains a NCS Software Programmer's Manual to document the procedure used in transferring, compiling, and otherwise using the software, and

- subject to the Software System Team Chairperson's approval, procures and maintains computer equipment to perform archiving and testing responsibilities.

E.1.5 Installation nuclear criticality safety organization. The installation nuclear criticality safety organization

- ensures all users of the NCS Software System utilize software that is covered by this Configuration Control Plan for mainframe computations,

- ensures the computer software contained in the Software Catalog (Form E.3) is properly validated for the intended use,

- assists in the performance of Verifications and Configuration Control tests, as necessary,

- authorizes access to the software covered under this plan for users in the installation CSO and other contractors per the User Access form (Form E.4), forwards completed User Access forms to the Software System Administrator, and provides notification to the Software System Administrator when user access needs to be removed,

- develops and implements Disaster Plans where appropriate and forwards a copy of these plans to the Software System Team, Form E.5,

- ensures that each user granted access to the software is provided with training in the proper use of the software,

- develops and implements the appropriate Quality Assurance and Quality Control Programs to ensure the correctness of calculational results and use of the software,

- assists the Software System Team in implementing software changes, testing new software, user access control, and any other areas where appropriate,

- may request changes by initiating the Software Revision Report (Form E.1) in order to define modification requirements, and

- reports problems encountered to the proper Functional System Manager using the Software Nonconformance Report (Form E.2).

E.1.6 Software developer. The software developer

- makes ONLY those software changes that have been approved by the Software System Team on a Software Revision Report (Form E.1),

- may propose software changes on a Software Revision Report,

- updates software version identification in a program when changes are made,

- assists the Software System Team in conducting the Verification Test of the software modification,

- supplies information to the System Administrator on software version identification and software changes, as appropriate, and

- works with Software System Team to update the supporting documentation.

E.2 Software identification. Initial system configuration consists of a catalog of application specific software. This Software Catalog defines the baseline system configuration. Access control is established by the Software System Team and is maintained by the System Administrator. Unambiguous labeling shall provide traceability from source modules to executable modules (Form E.6).

Versions shall be uniquely identified in such a way that the update sequence may be readily determined. The version number and revision number shall be listed at least once on all output.

E.3 Software control. Users of software are responsible for ensuring that any software used is the currently approved version and that the use and application is validated.

All modifications to the nuclear criticality safety software system require the approval of the Software System Team using the procedure in section E.4 of this plan.

The software residing in the Production Storage Area will be audited by the Quality Division to ensure that the correct version is in use and that no changes have been made.

Hard copy computer printouts shall have, printed on a header, the version and date of revision of the principal software unit generating the printout.

All modifications of software will be acceptance tested as specified on the Software Revision Report.

E.4 Software change procedure. A software change is initiated by any user by completing Part A of the Software Revision Report.

The request is sent to a member of the Software System Team.

The Software System Team Chair/Functional System Manager/Software Administrator transmits the report to the other members of the Software System Team, as needed, to determine if and when the change should be made.

Approval or rejection is documented by completing Part B of the Software Revision Report. If the modification is to be made, the Verification Test Plan shall be developed and documented on the Software Revision Report, Part B. NOTE: The level of detail in the Verification Test is determined by the Software System Team based on the extent of the software change and the consequences of unintended or unanticipated changes. The Software Revision and associated Verification Test are approved by the Software System Team by signing the appropriate spaces on the form. If the Software Revision Report is rejected, the Software System Team Chairperson provides an explanation for rejection and provides a copy to the requestor.

A copy of the approved Software Revision Report (Parts A and B) is sent to the System Administrator. The System Administrator provides the Software Developer a copy of the current source code.

The software modifications are made in the Development Storage Area. Once the software modifications have been made to the satisfaction of the Software Developer and the System Administrator, the software is transferred to the Migration Storage Area by the Functional System Manager. Part C of the Software Revision Report documents the completion of this step.

The Verification Test is performed in the Migration Storage Area by the Functional System Manager with assistance, where appropriate, from the installation NCS Organizations.

The performance of the software in the Verification Test is evaluated by the Software System Team. Part D of the Software Revision Report documents the Verification Test results and the acceptance/rejection of the results by the Software System Team.

Software System Team approval of the Software Revision Report, Part D, provides notification to the System Administrator to transfer the new version of the software into the Production Storage Area and a copy of the current version (source and executable code) to the Archive Storage Area.

Completion of Part E of the Software Revision Report documents the software transfers, bit-by-bit comparison of the new Production version, and completion of the software revision procedure.

E.5 Nonconformance Report procedure. A Nonconformance Report is initiated by completing Part A of the Nonconformance Report (Form E.2).

The request is sent to a member of the Software System Team.

The Software System Team Chair/Functional System Manager/Software Administrator transmit the report to the other members of the Software System Team, as needed, to determine the actions to be taken to prevent recurrence of the nonconformance.

The Software System Team Chair provides nonconformance notification to the Quality Assurance Division and the Occurrence Reporting System, as appropriate.

In extraordinary cases, the System Administrator or the Software System Team Chairman may authorize shutting down a program that presents immediate and major danger to safety or the environment. In such cases, the Software System Team shall authorize the use of the corrected software, full details of the incident shall be provided in the documentation for the change, and a Nonconformance Report shall be initiated. The changed software shall have a new version identification.

E.6 Software testing. Configuration Control Test: Testing procedure, requirements, and plan are determined by the Software System Team. At a minimum, the Configuration Control Test should include (a) a periodic (every quarter) bit-by-bit comparison of the production version against an archived production version stored at the time the production version was installed, and (b) quarterly testing by each installation using installation specific validation cases. Documented records of these tests shall be maintained by the System Administrator.

Verification Test: Testing procedures, requirements, and plans are determined by the Software System Team pursuant to section E.4 of this plan. The level of detail found in the test plan will be commensurate with the complexity of the software change. As part of a Software Change Request implementation, transfer tests will be performed to verify the copying and transferring of software from one computing platform to another computing platform as listed in the software catalogs.

Form E.1 Software Revision Report.

SAMPLE

| **Part A - Request for Software Change** (to be completed by Software User/Developer) | Report No. **SRR-** |
|---|---|
| Reason for the requested change and Software Nonconformance Report No.(**SNR-** ): | |
| Description of requested change: | |
| Modules affected: | |
| Describe anticipated or known effects the change will have on:<br>A. Sample problem results<br>B. Calculational time/efficiency<br>C. Existing documentation | |
| Name of requestor and signature: | Date: |

**Part B - Software System Team Approval/Rejection**
(to be completed by Software System Team)
(approval requires four affirmative signatures from Software System Team)

| | Approval | Rejection |
|---|---|---|
| Functional System Managers | | |
| | | |
| | | |
| System Administrator | | |
| Software System Team Chairperson | | |
| Reason for rejection: | | |
| Software Verification Test Plan attached?____ | | Page 1 of 2 |

Form E.1 (cont.)

## Part C - Software Change Documentation
(to be completed by Software Developer and System Administrator)

Describe the change and components affected

File names for new source or data:

Describe the results of the Software Developer testing performed:

Does the change affect existing documentation? If so, update and attach new documentation.

Software change completed
Software Developer _____ Date _____
System Administrator _____ Date _____

Software transfer: Development Storage Area to Migration Storage Area by Functional System Manager
SYS01 _____ Date _____ SYS03 _____ Date _____
SYS02 _____ Date _____ SYS04 _____ Date _____

## Part D - Software Verification Test Evaluation (verification results attached)
(to be completed by System Software Team)

Verification tests results accepted and permission granted to transfer software from Migration Storage Area to Production Area
Functional System Manager _____ Date _____
Functional System Manager _____ Date _____
Functional System Manager _____ Date _____
System Administrator _____ Date _____
Software System Team Chairperson _____ Date _____

## Part E - Software Change Implementation in Production Storage Area
(to be completed by System Administrator)

| Computer designator | Archive & load transfer date | Bit-by-bit compare date | Update procedure date | Functionality test date | Restore user access date | Update catalog date |
|---|---|---|---|---|---|---|
| SYS01 | | | | | | |
| SYS02 | | | | | | |
| SYS03 | | | | | | |
| SYS04 | | | | | | |

Software change implementation in Production Storage Area completed and updated software catalogs sent to Software System Team Chairperson.
System Administrator _____ Date _____

Form E.2 Software Nonconformance Report.

SAMPLE

| **Part A - Report of Software Nonconformance or Error:** (to be completed by Software user) | Report No. **SNR-** |
|---|---|
| Software user name and address: | |
| Software title/version/date: | |
| Description of software nonconformance or error: | |
| Cause of nonconformance or error: | |
| Effect on previous calculations: | |
| Recommended corrective action: | |
| **Part B - Software Nonconformance Assessment and Action Plan** (to be completed by Software System Team) | |
| Cause of nonconformance and effect on previous software users: | |
| Immediate action is required to stop use of software? _____ | |
| Reportable event per Occurrence Reporting System? _____ | |
| Recommended corrective action: | |
| Software System Team approval of recommended corrective actions:<br>Functional System Manager _____ Date _____<br>Functional System Manager _____ Date _____<br>Functional System Manager _____ Date _____<br>System Administrator _____ Date _____<br>Software System Team Chairperson _____ Date _____ | |
| Corrective actions completed<br>Software System Team Chairperson _____ Date _____ | |

Form E.3 NCS Software System Version No. 1 Catalog

Computer node _____

Updated: _____

SAMPLE

| Module | Date | Revision | Data set Name |
|---|---|---|---|
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

Form E.4 Request for User Access.

User access is requested for the following Contractor Nuclear Criticality Safety software:

The proposed user and their supervisor have been informed and understand that validation, (establishment of correctness or bias in calculated results) is a user responsibility and that the contractor makes no claim of correctness for the computer software or for computer calculations performed by others.

Type Proposed User's Name and UID:_____

Proposed User (Signature): _____ Date: _____

User's Address: _____ User's Phone #: _____

User's Supervisor (Signature): _____ Date:_____

Organization: _____

Installation Nuclear Criticality
Safety Organization Head (Signature): _____ Date:_____

SEND COMPLETED FORM TO:

_____

(TO BE COMPLETED BY SOFTWARE SYSTEM ADMINISTRATOR)

User access was activated on this date: _____

System Administrator Signature: _____

Copy: Software System Team Chair

Form E.5 NCS Software Disaster Plan.

A disaster plan is not necessary for the NCS software because of the redundancy provided by multiple computing systems. The NCS software will be provided on the following systems, for example:

1. Computing System #1 I.D.
2. Computing System #2 I.D.
3. Computing System #3 I.D.

Therefore, it is judged to be incredible that all NCS software versions could be simultaneously destroyed.

Form E.6 Software Labeling Protocol Examples

Source
NCSS.ZAZ39461.Module.V#R###.FORT (.ASM)

Production Subroutine library
NCSS.ZAZ39461.Sublib.V#R###.LOAD

Archive Subroutine library
NCSS.ZAZ39461.Sublib.V#R###.ARCHIVE

Production Load Modules
NCSS.ZAZ39461.module.V#R###.S###.LOAD

Migration Load Modules
NCSS.YCR39461.module.V#R###.S###.LOAD

Archive Load Modules
NCSS.ZAZ39461.module.V#R###.S###.ARCHIVE

Data Libraries
NCSS.ZAZ39461.identification.V#R###.DATA

MODULE = program name (such as KENOVA, CSAS25, and SUBLIB)

V# is the nuclear criticality safety software version number.
R### is the module revision number.
S### is the subroutine library revision number.

Form E.7 NCS Software System Team (NCSSST) Sample Charter.

Objective: The nuclear criticality safety software system team (NCSSST) acts as the change control board for the company's Nuclear Criticality Safety Software. The team should:

- maintain the company's Nuclear Criticality Safety Software Configuration Control Plan,
- determine and implement necessary changes to the NCS software pursuant to the configuration control plan,
- address NCS software nonconformance reports as appropriate, and
- provide assistance to other organizations in the area of software configuration control.

Mtng Freq: At the discretion of the team (minimum - once per year)

Team Membership:
Chairperson
Contractor Central Safety and Health organization or designee
System Administrator
Computer, hardware or software maintenance/operations organization
Installation Functional System Manager(s)
Installation representative(s)

Reporting: The NCSSST is directly accountable to the Contractor Central Safety and Health organization.